System Modeling combined with Dimensional Analysis for Conceptual design

François Christophe, Raivo Sell, Eric Coatanéa, Mart Tamre

System Modeling combined with Dimensional Analysis for Conceptual design

F. Christophe, R. Sell, E. Coatanéa and M. Tamre

Machine Design, Helsinki University of Technology, P.O.Box 4100, FIN-02015 HUT; Phone: +35894513578, Fax: +35894513549, e-mail: francois.christophe@tkk.fi

Department of Mechatronics, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia; Phone: +3726203201, Fax: +3726203203, e-mail: raivo@staff.ttu.ee

Machine Design, Helsinki University of Technology, P.O.Box 4100, FIN-02015 HUT; Phone: +35894516272, Fax: +35894513549, e-mail: eric.coatanea@tkk.fi

Department of Mechatronics, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia; Phone: +3726203202, Fax: +3726203203, e-mail: mart@staff.ttu.ee

Abstract— Early design is widely accepted inside the engineering design community as a crucial design phase. This is due to the fact that decisions taken at this stage constrained heavily the final performances of products. This article presents a design framework dedicated to the early design phases of mechatronic products. This framework provides a scientifically coherent methodology for refinement, analysis, modeling, comparison and evaluation of design solutions at early stage of the design process. SysML is proposed by the authors as a powerful modeling language properly adapted to mechatronic requirements. In addition, the article proposes to combine SysML with dimensional analysis and qualitative physics in order to provide a design tool able to proceed also with early simulations, comparisons and evaluations. The framework proposed in this article is based on strong theoretical bases and is the combination of two doctoral works. A third doctoral work is planned on the same research issue.

Index Terms— System modeling, Conceptual design, Dimensional Analysis, comparison of concepts, evaluation, unified design methodology, refinement, synthesis

I. INTRODUCTION

Early design process is a fundamental phase of the design process because it has been demonstrated by practitioners and scientists [1, 2] that 75% of the final cost of a product or service is constrained during the initial design phases due to decisions taken at this stage of the design process. The same analysis can be made for technical performances of a machine or device. Consequently, it is important for designers and people involved in the early design process to possess efficient modeling, comparison and evaluation tools. These tools should provide necessary insight in the modeling, evaluation and comparison of various types of performances of design concepts. In other terms, these tools should assist designers and other involved persons during the analysis and modeling stages. They should provide assistance and insight during the evaluation, comparison and improvement of early design concepts. At the moment, research in engineering design as provided a significant numbers of practical tools but most of them are focusing on the later design stages (i.e. embodiment and detail designs). Existing tools dedicated to analysis of design problems, modeling, evaluation and comparisons are characterized by the lack of commonly accepted fundamental scientific basis and definitions and by the poor repeatability of there results. These drawbacks have been pointed out by the scientific community [3]. The necessity of creating a coherent scientific synthesis for the early design process is an urgent matter in order to promote more scientific design approaches.

This article is an attempt to provide such type of synthesis of a coherent design approach combining analysis, evaluation and comparison of design concepts. The scope of this article is limited to mechatronic products but we argue that the real possibility of our approach is much broader and encompass other design aspects such as service and process design. Demonstrating this hypothesis has already been the goal of some of our research works but it requires still future investigations.

The present article is organized in the following manner. The second section is presenting basics of the SysML modeling approach. This language is an evolution of the UML modeling language and we aim at using it as a powerful tool for analyzing and modeling mechatronic design problems.

The third section presents a powerful methodology based on dimensional analysis and multi-agent optimization approach used for behavior simulation of machines and also for

comparing and evaluating concepts of solutions. The mathematical machinery provided by dimensional analysis can be fruitfully combined with the SysML modeling approach and provide a coherent framework for early design of mechatronic systems. This is a strong reason for us to propose this integrated framework.
The last section is a conclusion summarizing the results and presenting future and complementary research works.

II. CONCEPT MODELING WITH SYSML TOOLBOX

According to the International Council on Systems Engineering [4], the Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. The whole design process focuses on defining customer needs and requires functionality early in the development cycle, documenting requirements then proceeding with design synthesis and system validation while considering the complete problem of operations, performance, test, manufacturing, cost & schedule, training & support and disposal [4]. This definition points out the importance of early design and integrated activity very clearly setting high demands for modeling concepts and tools. Complex system design embraces several domains which have its own tools and techniques utilized for several years already.

In the software design world Unified Modeling Language (UML) is de facto standard for object-oriented software design. Started with UML 1.1 and UML 1.5 the most recent official version is now UML 2.0. The essence of software modeling (as in all modeling) is abstraction: the removal of fickle and distracting detail of implementation technologies as well as the use of concepts that allow more direct expression of phenomena in the problem domain [5]. One of the recent trends is the increasing software part in everyday products. According to this, there is increasing need for close communication and integration of techniques and tools between software design and conventional hardware design.

There are several attempts to apply UML for non-software design. Serious improvement has been reached in recent years. The important outcome is OMG SysML specification finalized this year (2007) which is initially derived from UML RFP: UML for System Engineers Request for Proposal [6] in 2003. However there is several state-of-art works carried out by research groups based on the UML profile mechanism.

- UML Profile for Schedulability, Performance, and Time Specification [7];
- UML 2.0 Profile for Embedded System Design [8];
- UML Testing Profile [9];
- UML Profile for SoC (Systems on Chip) [10];
- UML 2 to Solve Systems Engineering Problems [11];
- UML for Hybrid Systems [12].

Recently the major players in government, industry and ICT have collaborated to extend UML to cover the domain of Systems Engineering. This new standard - SysML is adopted by the Object Management Group in the autumn of 2005 [13]. During the 2007 finalized version of SysML 1.0 is expected. So far the version 1.0 draft from May 2006 is the adopted specification.

SysML reuses a subset of UML 2.0 diagrams and augments them with some new diagrams and modeling constructs appropriate for systems modeling. SysML is designed to complement UML 2.0, so systems engineers who are specifying a system with SysML can collaborate efficiently with software engineers who are defining a system with UML 2.0 [13]. Four pillars of SysML are shown figure 1.

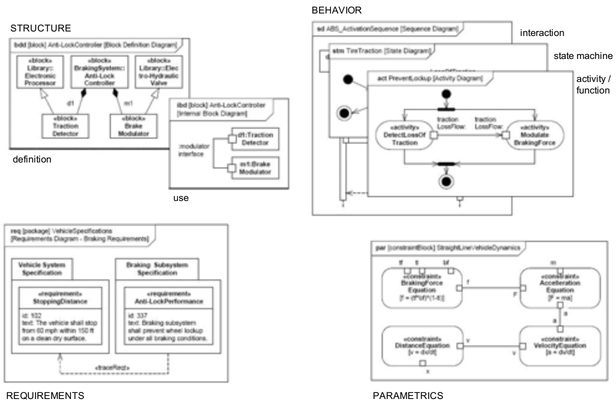

Figure 1 SysML pillars

In mechatronics and system engineering very wide range of applications can be considered. Different products and domains have its own specifics and therefore it is necessary to customize general system modeling tools to meet the specifics of particular application domain. On the same time the connection and compatibility have to remain. UML and SysML have the profiling mechanism to extend and/or restrict the initial language constructs ensuring the required compatibility on the same time. In this paper we introduce the SysML toolkit which consists of a SysML profile for mobile platform development in conceptual stage.
The toolkit is defined as a SysML profile and external simulation package. The profile itself consists of template libraries, diagram extensions and model libraries. Standard model libraries are *Principle*, *Terrain* and *ContactType*.
The model library *Principle* is a collection of standard mechatronics sub-systems, elements and working principles. This library is the most similar to the existing design software packages part library concept where standard parts are defined and collected into the categories. The Mobile Platform Toolkit (MPT) *Principle* library consists of the working principles and subsystems formulated in SysML and of the extended profile. This means that similar subsystems can be found in different libraries but the abstraction level is higher and the subsystem is defined in formal language rather than physical component. The boundaries between the physical domains are not defined very sharply and can be determined later at the detailed design stage. The model can be developed by linking the subsystems

and working principles from library with loosely coupled relations whereas the certain key parameters are defined. These parameters are in most cases derived from requirement model and are related with many other parameters of a system. The general structure of the toolkit is shown in figure 2. This figure presents the toolkit structure of a mobile robot platform.

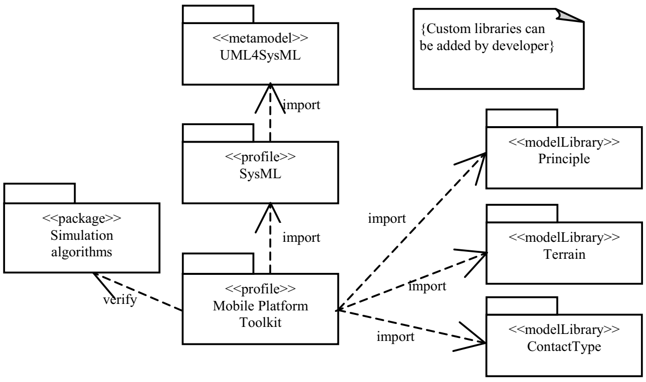

Figure 2 Mobile Platform Toolkit structure

Terrain and *ContactType* library are holding the parameters of different terrain and vehicle-soil contact. The reason for establishing the *Terrain* and *ContactType* library was the mobile platform performance analysis and simulation need. Depending on the required terrain capabilities, the mobile robot must deal with obstacles, surface characteristics, slopes, etc. Terrain properties have a great affect for robot design where the smart and optimal design can save the energy, improve the performance, optimize the budget and so on. These parameterized models can be linked to the design element or design candidate and used in initial simulations.
The conceptual modeling exploits several SysML defined diagrams with extended toolkit objects. Toolkit specified the modeling steps and appropriate diagrams according to the application. In figure 3 the system main services are modeled in Use Case diagram where MPT specific stereotypes are used.

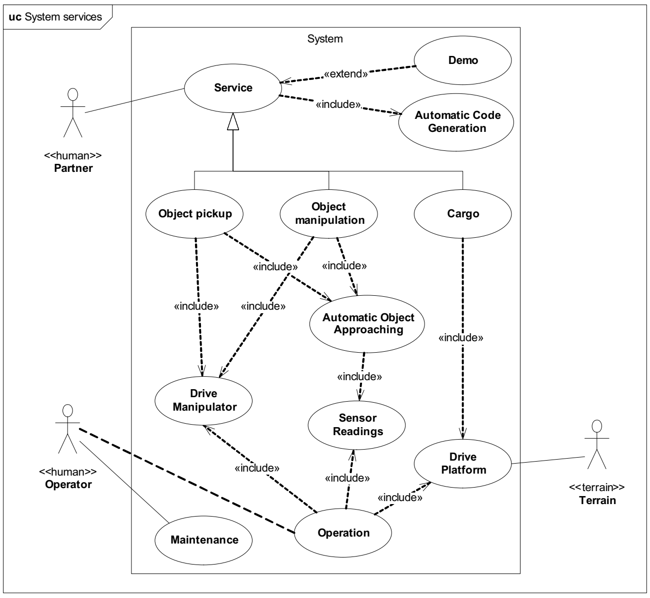

Figure 3 System services

For the structure and behavior similar diagram types are specified. The toolkit specification has been further studied in the following work of Sell [14].

The simulation is usually exploited on the later design stage where the system model is relatively precisely defined. To get the maximum benefit, the proposed design framework includes the simulation into the conceptual design stage. The model (structure and behavior) consists of special block element stereotypes as *simu*. An example is shown on figure 4 where *simu* block is a control algorithm of robot, controlling the leg and wheel motors according to terrain changes. The *ControlFPGA* block is a link to the simulation model shown in figure 4. Simulating the control algorithm, the engineering team gets the feedback of critical component parameters required to fulfill the initial requirements or simulating different algorithm candidates determining the system feedback. The following example shows the *ibd* containing *simu* element.

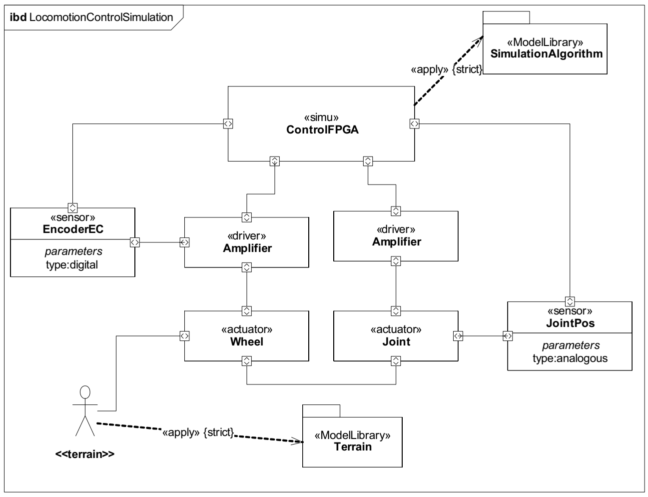

Figure 4 Simulation block in system structure diagram

Many non-traditional techniques and methods on engineering problem solving domain have been come to the fore recently. One of the reasons is definitely increase of the computing power. These opportunities allow solving the engineering tasks, which can not be described with linear differential equations and are non-deterministic. The techniques applicable for more advanced mechatronics system modeling, which are taken into account, are followings:

- Multi-Agent systems,
- Genetic Algorithms/ Genetic Programming,
- Neural Network Schemas,
- Fuzzy logic.

These methods are successfully applied in several cases for solving specific problem on optimization, machine learning, adaptive control, path planning, etc. field. For example fuzzy logic is widely used in controller systems or neural networks on parameter prediction. However in many cases the theory is applied only in computer environment while calculation or simulation certain problem. Genetic algorithms are often used for finding global optimum in case of great state space. The advantage of AI methods over the traditional is the ability to search over entire solution space and they are applicable to a wide range of problems including non-continuous functions and functions involving different types of variables. Although here are lots of research results and success exploiting AI techniques for a certain problem, applying the single technique for a complex interdisciplinary problem as the mechatronics concept generation definitely is applicable, is not a trivial task. Nevertheless there has been limited number of attempts of exploiting above techniques for a design solution generation. Some works [15, 16, 17] have shown the possibility to apply the multi-agent system, genetic programming and bond graph combination to automate the initial system concept generation. SysML modeling toolkit can be combined with a theoretical approach dedicated to early evaluation and comparison. This theoretical framework can provides a useful complement to the modeling approach in order to simulate qualitatively, compare and evaluate solutions. The theoretical basis is described in the following part and the practical combination of these approaches will be extensively investigated in future research works.

III. SIMULATION AND COMPARISON OF CONCEPTS WITH DIMENSIONAL ANALYSIS

A. Dimensional Analysis and behavior simulation

1) Basis of Dimensional Analysis

Dimensional Analysis (DA) is a field of Qualitative Physics which concerns units and magnitudes. DA is often used in order to verify the dimensional homogeneity of physical equations but its scope is much broader. Similarities between scales are major area of contribution [18]. The fields of application are numerous; we can quote electromagnetic theory, aerodynamics, aeronautic. DA mostly relies on the Vashy-Buckingham theorem which states that the study of a physical problem expressed with *n* dimensional quantities can be reduced by a factor k when expressed in a dimensionless form. Dimensionless numbers such as Reynolds and Froude are resulting from the DA method. Bashkar and Nigam have provided a machinery to allow the use of DA in the analysis of a mechanism [19]. This machinery provides powerful tool for the behavioral simulation of a mechanism. Furthermore, it has been proved in [20] that under certain conditions, there exists a formal link between the topological structure of design and the metric space provided by DA. Thus DA can be used in conceptual design for simulation and comparison purposes.

2) Dimensionless numbers computation

The Vashy-Buckingham theorem does not provide any specific guidance related to the choice of the variables used for the reduction of the problem. In order to enable systematic computation of dimensionless numbers, we consider the input and output variables of a concept as performance variables. Then the choice of repeating variables should be done within the concept's internal variables and according to the unique number of the system's governing dimensions.

This systematic computation can be done according to Butterfield's paradigm [21]. This paradigm is used in order to select the minimum set of repeated variables which ensures the non-singularity of the metrization procedure. This procedure provides one dimensionless group for each concept. The practical computation of dimensionless numbers is not developed in this short article but can be followed in a detailed manner in [20].

3) Simulation of the behaviors a concept

The simulation of the behaviors of a concept of solution is the immediate result of the Dimensionless Group computation. In fact, the dimensionless numbers computed for one concept allow us to qualitatively show the evolution of each variable according to the variation of the other variables [19].

As an example, we can consider an electrical battery and simulate is charging phase. In this example we consider the following variables: U potential of the battery, I its charging intensity, E the energy stored, Ω its internal resistance, ρ_V the volume density of the battery and ρ_M its mass density. For that device, the variables of interest are U and I, the other ones being internal variables. DA gives us two dimensionless numbers:

$$\pi_U = U.E^{-1}.\Omega^{-1/2}.\rho_V{}^{1/2}.\rho_M{}^{-1/4} \quad (1)$$

$$\pi_I = I.E^{1/3}.\Omega^{1/2}.\rho_V{}^{-5/6}.\rho_M{}^{-1/4} \quad (2)$$

From this dimensionless group, we can simulate the behavior of a certain type of battery during the charging phase, considering Ω, ρ_V and ρ_M as known.

Indeed if the battery is charging, U should increase. An increase of U implies an increase of the amount of energy stored E. From π_I, we can deduce that an increase of E will lead to a decrease of the intensity of charge I. This example efficiently reflects the normal behavior of a battery being charged. The simulation procedure can be generalized to any kind of complex mechanism and can explain qualitatively its physical behavior [20] [19]. This is a part of the theoretical background based on the principle of similarity which allows early simulations of complex mechanisms. The similarity principle can also be used for the purpose of comparison between concepts of solutions. This is the goal of the following section.

B. Principle of similarity and comparison of concepts of solutions

1) Similarity principle

In order to be comparable, two concepts of solution should share the same function and provide the same type of output variables. This means in practice that the dimensionless numbers involving the output variables should be equal regardless of the internal variables of the concepts. This is the similarity principle [22] [18].
The similarity principle can be expressed in the following manner if we consider two concepts *1* and *1'* sharing same type of variables.

$$\pi_1 = A^{\alpha} B^{\beta} C^{\chi} \ldots X^{\zeta} \quad (3)$$

If the scales of the parameters are varying from one machine to another, then we have:

$$\pi'_1 = \left(\frac{A}{m}\right)^{\alpha} \left(\frac{B}{n}\right)^{\beta} \left(\frac{C}{o}\right)^{\chi} \ldots \left(\frac{X}{p}\right)^{\zeta} \quad (4)$$

where *m, n, o* and *p* are the scales ratios.
In order to meet the similarity condition for the Pi number π_1 and $\pi_{1'}$, we need to fulfill the following condition which is resulting from Equations 3 and 4.

$$A^{\alpha} B^{\beta} C^{\chi} \ldots X^{\zeta} = \left(\frac{A}{m}\right)^{\alpha} \left(\frac{B}{n}\right)^{\beta} \left(\frac{C}{o}\right)^{\chi} \ldots \left(\frac{X}{p}\right)^{\zeta} \quad (5)$$

This means that, the similarity condition is:

$$m^{\alpha} n^{\beta} o^{\chi} \ldots p^{\zeta} = 1 \quad (6)$$

This principle presented here in a simple case can be generalized for concepts where Pi numbers are expressed using different type of variables. This is part of an ongoing research investigation.

2) Comparison method

In order to compare different concepts, we define an ideal concept (i.e. a usual approach used in multi-objective optimization) according to ideal target values of the performance variables. The comparison procedure can be done between the ideal concept and the real concepts respecting the principle of similarity. The aim is to define for a real concept, the real values of the performance variables both *approaching the ideal values* and *meeting the similarity principle*. This approach leads to a combinatorial optimization procedure. The complexity of this problem grows exponentially with the amount of performance variables.

3) Agent based optimization

Multi Agents Systems (MAS) have revealed to be very efficient in multi-objective optimization problems. The aim for us is to use them to tackle the exploration complexity involves by the optimization procedure described above. The agent based method, we propose in this research work, can be seen as a population of concepts trying to find optimal values for their variables according to performance constraints. We argue that this method allows us avoiding any kind of weighting approach commonly used in design and source of subjectivity in the selection and evaluation of concepts.
Indeed, each attribute of performance is supposed in a first step to have the same importance than the others. In a second time a level of importance can be given if needed. The multi-agent optimization procedure is part of current investigations and is a powerful method to explore the design space.

IV. CONCLUSION

This paper has presented the initial development of a synthetic approach for refining, creating and evaluating solutions during the early design process. The synthetic approach is dedicated in this work to mechatronic systems. The method relies on SysML modeling language used for modeling and refining the design problem, dimensional analysis and qualitative physics used for comparing, evaluating and simulating the solutions. The method is coherent scientifically and based on proved scientific concepts. The approach is aimed at guiding the designers from the validation of the needs to the comparison and evaluation of mechatronic solutions. The document itself is very general and is not describing in a very detailed manner the global approach. This is due to the length constrained of the article. The SysML language is still under development but has been developed specifically for systems modeling; in this respect this modeling language is suited for our purpose. In the same vein, dimensional analysis, qualitative physics principles and the extensive use of the concept of similarity is novel in the sense that it has never been used in a systematic manner for design purpose. This work should be viewed as an initial attempt to provide a complete early design framework for mechatronic systems. We need in future to develop a complete software environment to improve the usability of our methodology. This will be the goal of future research works. We argue that the approach developed in this document is coherent and can be fruitfully developed and expended in near future. The concepts behind this approach are rather simple but the theoretical background is already very developed and has already been tested using several types of design problems.

Acknowledgment

The work of Raivo Sell is supported by the Estonian Ministry of Science and Education grant 0142506s03. The work of Eric Coatanéa has been performed within the research project COMODE. The COMODE project has received research funding from the EIF Marie Curie Action, which is part of the European Community's Sixth Framework Program.

References

[1] B. Lotter, "*Manufacturing Assembly Handbook"*. Butterworths, Boston, 1986.
[2] W. Hsu, I.M.Y. Woon, "Current Research in the Conceptual Design of Mechanical Products", *Computer-Aided Design*, Vol. 30, No. 5, pp. 377-389, 1998.
[3] J. Blessing, Consolidation of design research: the issue of design theory, workshop "Engineering Design Science: Consolidation and perspective", *International Conference Design 2006*, Dubrovnik, 2006.
[4] "*Systems Engineering Handbook*", INCOSE-TP-2003-016-02, v 2a, Technical Board of International Council on Systems Engineering, 2004
[5] G. Booch, J. Rumbaugh, and I Jacobson, *"The Unified Modeling Language User Guide"*, Addison Wesley, MA, 1999.
[6] *"UML for Systems Engineering RFP"*, OMG document ad/03-03-41, 2003. http://syseng.omg.org/UML_for_SE_RFP.htm
[7] *"UML Profile for Schedulability, Performance, and Time Specification"*, OMG document ptc/2003-03-2, 2003
[8] P. Kukkala, J. Riihimäki, M. Hännikäinen, T. D. Hämäläinen, K. Kronlöf, "UML 2.0 Profile for Embedded System Design". in *Proc. of the Design, Automation and Test in Europe Conference*. Munich, 2005.
[9] *"UML 2.0 Testing Profile Specification"*, version 1.0, OMG document formal/05-07-07, 2005
[10] S. P. Rajan, T. Hasegawa, M. Shoji, Q. Zhu, and T. Nakata, "UML Profile for SoC RFC", DAC 2005 Workshop, *UML-SoC 2005 UML for SoC Design Conference*, Anaheim, 2005.
[11] A. Gurd, "Using UMLTM 2.0 to Solve Systems Engineering Problems". Telelogic, 2003. http://whitepapers.zdnet.co.uk
[12] K. Berkenkötter, S. Bisanz, U. Hannemann, J. Peleska , "HybridUML Profile for UML 2.0" in *International Journal on Software Tools for Technology Transfer*, Springer Berlin / Heidelberg, Vol 8, No 2, 2006
[13] *"System Modeling Language (SysML) Specification"*. Version 1.0 Draft. OMG document ad/2006-03-01, 2006. http://www.sysml.org
[14] R. Sell, "Integration of V-model and SysML for advanced mechatronics system design", *Proc. of Int. Workshop on Research & Education in Mechatronics*, France, 2005
[15] G. Rzevski, "On Conceptual design of intelligent mechatronic system". in *Mechatronics* vol. 13, 2003, pp.1029–1044.
[16] J. J. Granda, "The role of bond graph modeling and simulation in mechatronics systems An integrated software tool: CAMP-G, MATLAB–SIMULINK" in *Mechatronics* vol. 12, 2002, pp. 1271–1295.
[17] K. Seo, Z. Fan, J. Hu, E. D. Goodman, R. C. Rosenberg, "Toward a unified and automated design methodology for multi-domain dynamic systems using bond graphs and genetic programming" in *Mechatronics* vol. 13, 2003, pp. 851–885.
[18] A.A. Sonin, "The physical basis of dimensional analysis", 2nd edition, Department of Mechanical Engineering MIT Cambridge,MA 02139, 2001.
[19] R. Bhashkar, A. Nigam, "Qualitative physics using dimensional analysis", *Artificial Intelligence*, vol. 45, pp. 73-111, 1990.
[20] E. Coatanéa "*Conceptual Design of Life Cycle Design: A modeling and evaluation method based on analogies and dimensionless numbers",* ISBN 951-22-7852-9, Doctoral dissertation, Helsinki University of Technology, 2005.
[21] R. Butterfield, Proceedings of the I MECH E Part C *Journal of Mechanical Engineering Science*, Volume 215, Number 11, 23 November 2001, pp. 1365-1375(11), 2001.
[22] W. Matz, *"Le principe de similitude en Génie Chimique",* Dunod, Paris, 1959.